UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/772,902	02/05/2004	Peter C. Huene	MSFT-2929/303466.01	5355

41505   7590   05/19/2008
WOODCOCK WASHBURN LLP (MICROSOFT CORPORATION)
CIRA CENTRE, 12TH FLOOR
2929 ARCH STREET
PHILADELPHIA, PA 19104-2891

EXAMINER
NAHAR, QAMRUN

ART UNIT	PAPER NUMBER
2191	

MAIL DATE	DELIVERY MODE
05/19/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

PTOL-90A (Rev. 04/07)

	Application No.	Applicant(s)
Office Action Summary	10/772,902	HUENE ET AL.
	Examiner	**Art Unit**	
	QAMRUN NAHAR	2191	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1) ☒ Responsive to communication(s) filed on <u>07 April 2008</u>.
2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4) ☒ Claim(s) <u>1-27</u> is/are pending in the application.
    4a) Of the above claim(s) _____ is/are withdrawn from consideration.
5) ☐ Claim(s) _____ is/are allowed.
6) ☒ Claim(s) <u>1-27</u> is/are rejected.
7) ☐ Claim(s) _____ is/are objected to.
8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9) ☐ The specification is objected to by the Examiner.
10) ☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.
    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
    a)☐ All   b)☐ Some * c)☐ None of:
      1.☐ Certified copies of the priority documents have been received.
      2.☐ Certified copies of the priority documents have been received in Application No. _____.
      3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08)
    Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413)
    Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1.   This action is in response to the RCE/amendment filed on 04/07/2008.

2.   Claims 1-27 are pending.

Response to Amendment

Claim Rejections - 35 USC § 102

3.   The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the

basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

4.   Claims 13-14, 17-22 and 24-27 are rejected under 35 U.S.C. 102(b) as being anticipated

by O'Leary (U.S. 5,950,000).

__Per Claim 13:__

The O'Leary patent discloses:

- associating at least one dynamic property with a build rule object associated with the

build tool, the dynamic property associated with a switch property; associating a value with

the switch property ("… When a file is selected, the IPE manager invokes Visual and specifies

the selected file as a command line parameter. … the name of the selected file …" in column 6,

lines 3-25; the name of the selected file is analogous to a switch property)

- and transforming a generalized command line comprising the name of the tool and at least one tag into an executable command line by programmatically replacing each tag in the generalized command line with an associated value ("… after the GUI builder 214 generates a makefile. The Build entry on the menu bar of GUI 300 is pulled down to disclose the target of make "app in /home/pat/src/Makefile" as the top reference on the build target picklist …" in column 6, lines 46-65).

Per Claim 14:

The O'Leary patent discloses:

- further comprising receiving a tool file describing the build tool, the tool file including the generalized command line and a build rule for transforming the generalized command line into the executable command line for executing the build tool (column 6, lines 3-25).

Per Claim 17:

The O'Leary patent discloses:

- further comprising generating the build rule object from the build rule (column 6, lines 38-45).

Per Claim 18:

The O'Leary patent discloses:

- wherein the build rule object generated from the build rule creates a dynamic property

descriptor (column 6, lines 3-25).

Per Claim 19:

The O'Leary patent discloses:

- wherein the value is stored in a generic property store (column 6, lines 18-23).

Per Claim 20:

The O'Leary patent discloses:

- wherein the value is associated with a particular use of a build rule in a project (column 6,

lines 3-25).

Per Claim 21:

The O'Leary patent discloses:

- wherein the value is associated with the switch property via a user interface (column 6,

lines 3-25).

Per Claim 22:

The O'Leary patent discloses:

- wherein the value is associated with the switch property via a scripting language (column 6, lines 3-25).

Per Claim 24:

The O'Leary patent discloses:

- receiving a file describing the build tool, the file including a build rule, the build rule comprising a generalized command line further comprising the name of the tool to be executed and at least one property associated with the tool and a rule for transforming the generalized command line into an executable command line for executing the tool (" ... When a file is selected, the IPE manager invokes Visual and specifies the selected file as a command line parameter. ..." in column 6, lines 3-25; "... load the file ... the "Generate Makefile" feature shown in FIG. 4 in the GUI 420 of the GUI Builder 214 is invoked to create a makefile for the application ... after the GUI builder 214 generates a makefile. The Build entry on the menu bar of GUI 300 is pulled down to disclose the target of make "app in /home/pat/src/Makefile" as the top reference on the build target picklist ..." in column 6, lines 38-65)

- generating a build rule object from the build rule by adding properties associated with the tool to said build rule object and setting the specified values on the build rule object

that is to be output ("… By selecting the menu item 502, IPE manager starts Visual and

commands the previous day Visual to **load the file** "app.xd" from directory "/home/pat/src."

After *adding some callbacks to the menu items and buttons*, the "Generate Makefile" feature

shown in FIG. 4 in the GUI 420 of the GUI builder 214 is invoked to create a makefile for the

application. The makefile is saved as "/home/pat/src/Makefile". …" emphasis added, in column

6, lines 38-45)

- associating a dynamic property with the build rule object, the dynamic property

associated with a switch property; associating a value with the switch property ("… When a

file is selected, the IPE manager invokes Visual and specifies the selected file as a command line

parameter. … the name of the selected file …" in column 6, lines 3-25; the name of the selected

file is analogous to a switch property)

- and transforming the generalized command line comprising the name of the tool and at

least one tag into the executable command line by programmatically replacing each tag in

the generalized command line with an associated value of the switch property ("… after the

GUI builder 214 generates a makefile. The Build entry on the menu bar of GUI 300 is pulled

down to disclose the target of make "app in /home/pat/src/Makefile" as the top reference on the

build target picklist …" in column 6, lines 46-65).

Per Claim 25:

      The O'Leary patent discloses:

- receiving a file describing a build tool, the file including a build rule, the build rule

comprising a generalized command line and a rule for transforming the generalized

command line into an executable command line for executing the tool ("… When a file is

selected, the IPE manager invokes Visual and specifies the selected file as a command line

parameter. … the name of the selected file …" in column 6, lines 3-25; "… load the file … the

"Generate Makefile" feature shown in FIG. 4 in the GUI 420 of the GUI Builder 214 is invoked

to create a makefile for the application … after the GUI builder 214 generates a makefile. The

Build entry on the menu bar of GUI 300 is pulled down to disclose the target of make "app in

/home/pat/src/Makefile" as the top reference on the build target picklist …" in column 6, lines

38-65).

Per Claim 26:

The O'Leary patent discloses:

- generating a build rule object from the build rule ("… load the file … the "Generate

Makefile" feature shown in FIG. 4 in the GUI 420 of the GUI Builder 214 is invoked to create a

makefile for the application …" in column 6, lines 38-45).

Per Claim 27:

The O'Leary patent discloses:

- associating a dynamic property with the build rule object, the dynamic property

associated with a switch property; associating a value with the switch property; and

transforming the generalized command line into the executable command line by

programmatically replacing a tag in the generalized command line with the value of the

switch property ("… When a file is selected, the IPE manager invokes Visual and specifies the

selected file as a command line parameter. …" in column 6, lines 3-25; and "… after the GUI

builder 214 generates a makefile. The Build entry on the menu bar of GUI 300 is pulled down to

disclose the target of make "app in /home/pat/src/Makefile" as the top reference on the build

target picklist …" in column 6, lines 46-65).

Claim Rejections - 35 USC § 103

5.   The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in
> section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are
> such that the subject matter as a whole would have been obvious at the time the invention was made to a person
> having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the
> manner in which the invention was made.

6.   Claims 1-2 and 7-12 are rejected under 35 U.S.C. 103(a) as being unpatentable over

O'Leary (U.S. 5,950,000) in view of Mason (U.S. 6,817,005).

Per Claim 1:

O'Leary teaches a processor ("… a central processing unit (CPU) 106 …" in column 3,

lines 34-36) with a dynamic property descriptor associated with a custom build rule object, the

dynamic property descriptor storing information associated with a property of the custom build

rule object, the property of the custom build rule object associated with at least one value ("… When a file is selected, the IPE manager invokes Visual and specifies the selected file as a command line parameter. … the name of the selected file …" in column 6, lines 3-25; the name of the selected file is analogous to a property of the custom build rule object); and a generic property store for storing the at least one value for the property of the custom build rule object ("… the name of the selected file is broadcast in a message … update their picklists …" in column 6, lines 18-23); the tool generating an executable command line comprising the name of the tool to be executed and one or more properties associated with the tool ("… after the GUI builder 214 generates a makefile. The Build entry on the menu bar of GUI 300 is pulled down to disclose the target of make "app in /home/pat/src/Makefile" as the top reference on the build target picklist …" in column 6, lines 46-65). O'Leary does not explicitly teach wherein said property comprises a command line switch.

However, Mason teaches wherein said property comprises a command line switch (column 21, lines 12-14).

It would have been obvious to one having ordinary skill in the computer art at the time of the invention was made to modify the computer system disclosed by O'Leary to include wherein said property comprises a command line switch using the teaching of Mason. The modification would be obvious because one of ordinary skill in the art would be motivated to efficiently integrate modules that are implemented separately (Mason, abstract).

Per Claim 2:

O'Leary further teaches further comprising a content handler, the content handler receiving a tool file, the tool file comprising at least one custom build rule and generating from the at least one custom build rule the custom build rule object ("… IPE manager …" in column 6, lines 3-25 and 38-65).

Per Claim 7:

O'Leary further teaches further comprising a dialog for adding or modifying the tool file (column 6, lines 10-13).

Per Claim 8:

O'Leary further teaches further comprising a dialog for adding or modifying the custom build rule (column 6, lines 57-59).

Per Claim 9:

O'Leary further teaches further comprising a dialog for adding or modifying the property of the custom build rule object (column 6, lines 57-59).

Per Claim 10:

O'Leary further teaches further comprising a dialog for adding or modifying the at least one value associated with the property of the custom build rule object (column 6, lines 10-19).

Per Claim 11:

O'Leary further teaches wherein the at least one value comprises a parameter value for the property of the custom build rule object ("… When a file is selected, the IPE manager invokes Visual and specifies the selected file as a command line parameter. …" in column 6, lines 3-25).

Per Claim 12:

O'Leary further teaches wherein the custom build rule object transforms a generalized command line by programmatically replacing a tag with a property value to generate the executable command line (column 6, lines 46-65).

7.   Claims 3-6, 15-16 and 23 are rejected under 35 U.S.C. 103(a) as being unpatentable over O'Leary (U.S. 5,950,000) in view of Mason (U.S. 6,817,005), and further in view of Odaka (U.S. 2003/0140333).

Per Claim 3:

The rejection of claim 2 is incorporated, and further, O'Leary does not explicitly teach wherein the tool file is associated with a schema. Odaka teaches wherein the tool file is associated with a schema (pg. 2, par. 25, lines 7-13).

It would have been obvious to one having ordinary skill in the computer art at the time of the invention was made to modify the system disclosed by O'Leary to include wherein the tool file is associated with a schema using the teaching of Odaka. The modification would be obvious because one of ordinary skill in the art would be motivated to create customized tags

that offer great flexibility in organizing and presenting information (Odaka, pg. 1, par. 13, lines 1-5).

Per Claim 4:

The rejection of claim 2 is incorporated, and further, O'Leary does not explicitly teach wherein the tool file comprises an XML file. Odaka teaches wherein the tool file comprises an XML file (pg. 2, par. 25, lines 1-6).

It would have been obvious to one having ordinary skill in the computer art at the time of the invention was made to modify the system disclosed by O'Leary to include wherein the tool file comprises an XML file using the teaching of Odaka. The modification would be obvious because one of ordinary skill in the art would be motivated to create customized tags that offer great flexibility in organizing and presenting information (Odaka, pg. 1, par. 13, lines 1-5).

Per Claim 5:

The rejection of claim 4 is incorporated, and Odaka further teaches wherein the XML file is associated with an XML schema (pg. 2, par. 25, lines 1-13).

Per Claim 6:

The rejection of claim 5 is incorporated, and Odaka further teaches wherein the XML file is validated against the XML schema (pg. 2, par. 25, lines 1-13).

Per Claims 15 & 16:

These are method versions of the claimed system discussed above (claims 4-6), wherein

all claim limitations also have been addressed and/or covered in cited areas as set forth above.

Thus, accordingly, these claims are also obvious.

Per Claim 23:

The rejection of claim 15 is incorporated, and O'Leary further teaches wherein the file is

received by a content handler, the content handler generating from the file at least one custom

build rule object ("… IPE manager …" in column 6, lines 3-25 and 38-65). O'Leary does not

explicitly teach that the file is an XML file. Odaka teaches that the file is an XML file (pg. 2,

par. 25, lines 1-6).

It would have been obvious to one having ordinary skill in the computer art at the time of

the invention was made to modify the method disclosed by O'Leary to include that the file is an

XML file using the teaching of Odaka. The modification would be obvious because one of

ordinary skill in the art would be motivated to create customized tags that offer great flexibility

in organizing and presenting information (Odaka, pg. 1, par. 13, lines 1-5).

Response to Arguments

8.   Applicant's arguments with respect to claims 1-27 have been considered but are moot in

view of the new ground(s) of rejection.

In the remarks, the applicant argues that:

a)   O'Leary fails to teach the tool generating an executable command line comprising the

name of the tool to be executed and one or more properties associated with the tool as recited in

claim 1.

Examiner's response:

a)   The Examiner notes that claim 1 is currently rejected under O'Leary in view of Mason.

O'Leary teaches the limitation "the tool generating an executable command line comprising the

name of the tool to be executed and one or more properties associated with the tool ("... after the

GUI builder 214 generates a makefile. The Build entry on the menu bar of GUI 300 is pulled

down to disclose the target of make "**app in /home/pat/src/Makefile**" as the top reference on the

build target picklist ..." in column 6, lines 46-65; app in /home/pat/src/**Makefile** contains the

name of the tool "Makefile"; and "home", "pat", "src" are analogous to the properties associated

with the tool).

In the remarks, the applicant argues that:

b)   O'Leary fails to teach "the custom build rule object transforms a generalized command

line by programmatically replacing a tag with a property value to generate the executable

command line" as recited in claim 12 and "transforming the generalized command line into the

executable command line by programmatically replacing a tag in the generalized command line

with the value of the switch property" as recited in claim 27.

Examiner's response:

b)   O'Leary teaches wherein the custom build rule object transforms a generalized command line by programmatically replacing a tag with a property value to generate the executable command line (column 6, lines 46-65).

Further, O'Leary teaches transforming the generalized command line into the executable command line by programmatically replacing a tag in the generalized command line with the value of the switch property ("... When a file is selected, the IPE manager invokes Visual and *specifies the selected file* as a command line parameter. ..." emphasis added, in column 6, lines 3-25; *the selected file* is analogous to the tag; where *specifies the selected file* is analogous to replacing a tag in the generalized command line with the value of the switch property; and "... after the GUI builder 214 generates a makefile. The Build entry on the menu bar of GUI 300 is pulled down to disclose the target of make "app in /home/pat/src/Makefile" as the top reference on the build target picklist ..." in column 6, lines 46-65).

In the remarks, the applicant argues that:

c)   O'Leary fails to teach "transforming a generalized command line comprising the name of the tool and at least one tag into an executable command line by programmatically replacing each tag in the generalized command line with an associated value" as recited in claim 13; and further fails to teach "generating a build rule object from the build rule by adding properties associated with the tool to said build rule object and setting the specified values on the build rule object that is to be output" as recited in claim 24.

Examiner's response:

c)   O'Leary teaches transforming a generalized command line comprising the name of the

tool and at least one tag into an executable command line by programmatically replacing each

tag in the generalized command line with an associated value ("… When a file is selected, the

IPE manager invokes Visual and *specifies the selected file* as a command line parameter. …"

emphasis added, in column 6, lines 3-25; *the selected file* is analogous to the tag; where *specifies*

the selected file is analogous to replacing each tag in the generalized command line with an

associated value; and "… after the GUI builder 214 generates a makefile. The Build entry on the

menu bar of GUI 300 is pulled down to disclose the target of make "**app in**

/home/pat/src/Makefile" as the top reference on the build target picklist …" in column 6, lines

46-65; app in /home/pat/src/**Makefile** contains the name of the tool "Makefile") as recited in

claim 13.

Furthermore, O'Leary teaches generating a build rule object from the build rule by

adding properties associated with the tool to said build rule object and setting the specified

values on the build rule object that is to be output ("… By selecting the menu item 502, IPE

manager starts Visual and commands the previous day Visual to **load the file** "app.xd" from

directory "/home/pat/src." After *adding some callbacks to the menu items and buttons*, the

"Generate Makefile" feature shown in FIG. 4 in the GUI 420 of the GUI builder 214 is invoked

to create a makefile for the application. The makefile is saved as "/home/pat/src/Makefile". …"

emphasis added, in column 6, lines 38-45) as recited in claim 24.

In the remarks, the applicant argues that:

d)   Claims 3-6, 15-16 and 23 are rejected under 35 U.S.C. 103(a) as being unpatentable over

O'Leary (US 5,950,000) in view of Odaka (US 2003/0140333). The O'Leary reference relates to

a GUI interface for programmers to use in building static makefiles for integrating software

tools. The Odaka published application relates to a conditional compile environment for use in

web servers and web browsers.

The Odaka reference does not remedy the shortcomings of the O'Leary reference as

discussed above and, thus, the combination of O'Leary and Odaka fails to provide the teachings

needed to establish that claims 3-6, 15-16 and 23 are obvious. These claims are allowable for at

least the reasons given above. Accordingly, reconsideration and allowance are respectfully

requested.

Examiner's response:

d)   In response to applicant's arguments against the references individually, one cannot show

nonobviousness by attacking references individually where the rejections are based on

combinations of references. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981); *In re*

Merck & Co., 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986).

Conclusion

9.   Any inquiry concerning this communication from the examiner should be directed to

Qamrun Nahar whose telephone number is (571) 272-3730. The examiner can normally be

reached on Mondays through Thursdays from 9:00 AM to 7:30 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Wei Y Zhen, can be reached on (571) 272-3708. The fax phone number for the organization where this application or processing is assigned is (571) 273-8300.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the TC 2100 Group receptionist whose telephone number is 571-272-2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

/Qamrun Nahar/
Qamrun Nahar
May 19, 2008